





# REDUCTION PROCESS FOR KNOWLEDGE DATA GENERATION SIMULATIONS

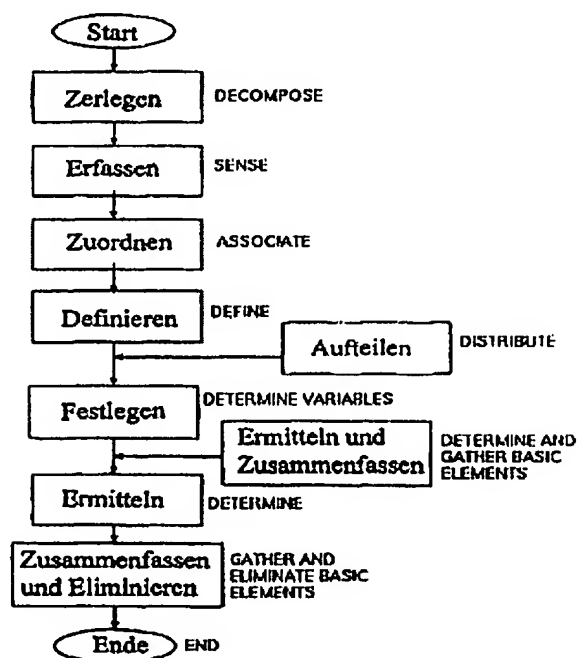
**Patent number:** EP1018087  
**Publication date:** 2000-07-12  
**Inventor:** HEINZELMANN ANDREAS (DE); REHFUS BERND (DE); BURKHARDT RAINER (DE)  
**Applicant:** DAIMLER CHRYSLER AG (DE)  
**Classification:**  
 - international: G06F17/50  
 - european:  
**Application number:** EP19980948855 19980821  
**Priority number(s):** DE19971042450 19970926; WO1998EP05325 19980821

Also published as:

 WO9917238 (A1)  
 WO9917238 (A1)  
 EP1018087 (A1)  
 DE19742450 (A1)

Abstract not available for EP1018087  
 Abstract of correspondent: **DE19742450**

A process for reducing the number of simulation steps required in a simulation process can be used for many different simulation processes for generating, by means of a computer, knowledge data on a total system composed of several electrically controlled elements. The elements are decomposed into components which comprise one or several basic elements. The electrical connections between the basic elements and between the components are then detected, discrete electrical state values are associated to the basic elements, and operating states, as well as possible states of the individual components, are defined in relation to the electrical state values of the basic elements. The variables which can be measured at the basic elements and components and are required to generate knowledge data are then determined and the basic elements which have no influence on the thus determined variables are determined, gathered and eliminated. The number of required simulation steps and the computing time required to carry them out can thus be reduced by several powers of ten.



Data supplied from the esp@cenet database - Worldwide





19 BUNDESREPUBLIK  
DEUTSCHLAND



DEUTSCHES  
PATENT- UND  
MARKENAMT

12 **Offenlegungsschrift**  
10 **DE 197 42 450 A 1**

51 Int. Cl.<sup>6</sup>:  
**G 06 F 17/50**

21 Aktenzeichen: 197 42 450.3  
22 Anmeldetag: 26. 9. 97  
43 Offenlegungstag: 8. 4. 99

DE 197 42 450 A 1

71 Anmelder:  
DaimlerChrysler AG, 70567 Stuttgart, DE  
  
72 Erfinder:  
Antrag auf Teilnichtnennung  
Heinzelmann, Andreas, 79771 Klettgau, DE;  
Burkhardt, Rainer, 71229 Leonberg, DE

56 Entgegenhaltungen:  
WEISANG, C., VOLLMAR, G.: Modellbasierte Wissensakquisition bei komplexen technischen Systemen - dargestellt am Beispiel der Kraftwerksführung, in: VDI Berichte Nr. 903, 1991, S. 111-124;  
ZEIGLER, B.P., u.a.: A Knowledge-Based Simulation Environment for Hierarchical Flexible Manufacturing, in: IEEE Trans. Systems, Man, Cybernetics - Part A, Vol. 26, No. 1, Jan. 1996, S. 81-90;  
HAREL, D., KRISHNAMURTHY, B.: A Graph Compaction Approach to Fault Simulation, in: Proc. of the 25 th ACM/IEEE Design Automation Conference, 1988, S. 601-604;  
HEILANDt, T., KRUCK, P.: Ein algorithmisches Verfahren zur Bewertung und Verdichtung von Entity-Relationship-Modellen, in: Informatik Forsch. Entw. (1993) 8, S. 197-206;

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

Prüfungsantrag gem. § 44 PatG ist gestellt

54 Reduktionsverfahren für Simulationen zur Wissensdatenerzeugung

57 Die Erfindung betrifft ein Verfahren zur Reduzierung der Anzahl der erforderlichen Simulationsschritte in einem Simulationsverfahren. Das Verfahren ist für viele verschiedene Simulationsverfahren verwendbar, die zur rechnergesteuerten Erzeugung von Wissensdaten über ein Gesamtsystem, bestehend aus mehreren elektrisch ansteuerbaren Bauteilen, eingesetzt werden. Die Bauteile werden in Komponenten zerlegt, die einen oder mehrere Grundbausteine umfassen. Danach werden die elektrischen Verbindungen zwischen den Grundbausteinen und zwischen den Komponenten erfaßt, den Grundbausteinen diskrete elektrische Zustandswerte zugeordnet und Betriebszustände sowie mögliche Komponenten Zustände der einzelnen Komponenten in Relation zu den elektrischen Zustandswerten der Grundbausteine definiert. Danach werden die für die Wissensdatenerzeugung notwendigen und an den Grundbausteinen und Komponenten meßbaren Größen festgelegt und jene Grundbausteine, die auf die festgelegten Größen keinen Einfluß haben, ermittelt, zusammengefaßt und eliminiert. Die Anzahl der erforderlichen Simulationsschritte und der zu deren Ausführung erforderlichen Aufwand an Rechenzeit lassen sich um mehrere Zehnerpotenzen reduzieren.

DE 197 42 450 A 1

## Beschreibung

Die Erfindung betrifft ein Verfahren zur Reduzierung der Anzahl der erforderlichen Simulationsschritte in einem Simulationsverfahren zur rechnergesteuerten Erzeugung von Wissensdaten über ein aus mehreren elektrisch ansteuerbaren Bauteilen bestehendes Gesamtsystem. Desweiteren betrifft die Erfindung einen Rechner zur Durchführung eines Reduktionsverfahrens.

Für eine Vielzahl von Anwendungen von aus elektrisch ansteuerbaren Bauteilen bestehenden Gesamtsystemen aus den verschiedensten Bereichen wie z. B. der Steuerung von Fertigungsstraßen oder Hochregallagern, der Robotertechnik, der Aufzugssteuerung, der Schaltkreisentwicklung, der Fahrzeugelektronik oder der dafür benötigten Diagnosesysteme ist es notwendig, das Normal- und Fehlverhalten der einzelnen Bauteile, die Auswirkungen dieses Verhaltens auf andere Bauteile und insbesondere auf das Gesamtsystem zu kennen.

Durch Datenmaterial über die Gesamtsysteme, z. B. aus deren Entwicklungsphase, ist es möglich, das Verhalten der Bauteile mittels Modellen zu beschreiben und durch Simulation Wissensdaten über das Verhalten der einzelnen Bauteile rechnergesteuert zu erzeugen. Bekannte Simulationsverfahren erlauben zwar die Nachbildung von einzelnen Bauteilen und Teilsystemen, scheitern aber sehr oft, wenn die Simulationen auf das Gesamtsystem ausgedehnt werden sollen. Dies ist unumgänglich, wenn die Auswirkungen des Einzelbauteilverhaltens oder des Verhaltens von Teilgruppen auf das Gesamtsystem untersucht werden sollen.

Dazu sind unzählige Simulationsdurchläufe notwendig, die immense Rechnerkapazitäten erfordern und zudem sehr zeitaufwendig sind.

Das US Patent 5,625,578 beschreibt ein Kontrollverfahren für Simulationen, die zur Untersuchung des elektromagnetischen Verhaltens von elektrischen Bauteilen einer gedruckten Schaltkarte eingesetzt werden. Die Simulationen basieren auf einem dem Schaltkreismuster der Schaltkarte äquivalenten Modell. Die Anzahl der äquivalenten Schaltkreiskomponenten wird um vier Größenordnungen reduziert, ohne Verluste hinsichtlich der Genauigkeit des Äquivalenzmodells nach sich zu ziehen. Das Schaltkreismuster wird durch geometrische Elemente dargestellt. Für das Äquivalenzmodell werden Gruppen von Hauptelementen zusammengestellt, die nur solche ausgewählte Elemente enthalten, deren Abstand zu den Nachbarelementen eine definierte Mindestwellenlänge  $\lambda$  überschreitet. Bei der Aufstellung von Matrixgleichungen, die die Maxwell'schen Gleichungen repräsentieren, werden den ausgewählten Elementen Feldwerte zugeordnet. Die Feldwerte der nicht berücksichtigten Elemente werden über die Feldwerte der ausgewählten Elemente definiert und über ein Rankingverfahren, das die Anzahl der ausgewählten Elemente berücksichtigt, wird eine das Äquivalenzmodell darstellende Matrix generiert.

Aufgabe der Erfindung ist es, ein Reduktionsverfahren für Simulationen zur Wissensdatenerzeugung bereitzustellen, das die Anzahl der erforderlichen Simulationsschritte erheblich reduziert, ohne daß für die Wissensdatenerzeugung relevante Informationen unberücksichtigt bleiben.

Diese Aufgabe wird erfindungsgemäß gelöst durch das in Anspruch 1 beschriebene Verfahren und durch den Rechner mit den Merkmalen des Anspruchs 12.

Das Verfahren reduziert die Anzahl der erforderlichen Simulationsschritte in Simulationsverfahren, die eingesetzt werden zur rechnergesteuerten Erzeugung von Wissensdaten über Gesamtsysteme, die aus mehreren elektrisch ansteuerbaren Bauteilen bestehen. Die elektrisch ansteuerbaren Bauteile werden in Komponenten zerlegt, die einen oder mehrere Grundbausteine umfassen. Danach werden die elektrischen Verbindungen zwischen den Grundbausteinen und zwischen den Komponenten erfaßt, den Grundbausteinen diskrete elektrische Zustandswerte zugeordnet und Betriebszustände sowie mögliche Komponentenzustände der einzelnen Komponenten in Relation zu den elektrischen Zustandswerten der zu einer Komponente gehörenden Grundbausteine definiert.

Im Anschluß daran werden die für die Wissensdatenerzeugung notwendigen und an den Grundbausteinen und Komponenten meßbaren Größen festgelegt und jene Grundbausteine ermittelt, die auf die festgelegten meßbaren Größen keinen Einfluß haben. Schließlich werden die einflußlosen Grundbausteine zusammengefaßt und eliminiert.

Mit diesem Verfahren läßt sich die Anzahl der erforderlichen Simulationsschritte und der zu deren Ausführung erforderliche Aufwand an Rechenzeit um mehrere Zehnerpotenzen reduzieren. Das Verfahren ist für viele verschiedene Simulationsverfahren verwendbar, die zur rechnergesteuerten Erzeugung von Wissensdaten über ein Gesamtsystem bestehend aus mehreren elektrisch ansteuerbaren Bauteilen eingesetzt werden.

Die Aufteilung der Wissensdaten in Wissensdatentypen ermöglicht die Auswahl eines oder mehrerer Wissensdatentypen, die für eine konkrete Fragestellung relevant sind. Dies ist vorteilhaft, da die an den Grundbausteinen und Komponenten meßbaren Größen nur für die ausgewählten Wissensdatentypen festzulegen sind und die Anzahl der erforderlichen Simulationsschritte nochmals reduziert wird.

Besonders vorteilhaft ist das Zusammenfassen von zuvor ermittelten Grundbausteinen, die auf denselben räumlich begrenzten Wissensdatenbereich Einfluß nehmen. Die Bildung lokaler Grundcluster berücksichtigt die Kenntnisse über die Bauteile und das Gesamtsystem, indem die Erkenntnis eingebracht wird, daß sich nicht alle Betriebs- und Komponentenzustände auf alle Wissensdaten auswirken. Es existieren räumliche begrenzte Wissensdatenbereiche, in denen sich nur bestimmte Komponentenzustände auswirken. Das Ermitteln und Zusammenfassen der jeweils betroffenen lokal einflußnehmenden Grundbausteine zu Grundclustern ermöglichen es, eine zunächst gesamtsystemumfassende Simulation auf die wesentlichen Simulationsschritte zu reduzieren und eine Eingrenzung der globalen Simulation auf einen begrenzten Wissensdatenbereich zu erzielen.

Weitere vorteilhafte Ausgestaltungen und Weiterbildungen des erfindungsgemäßen Verfahrens und des erfindungsgemäßen Rechners sind in den Unteransprüchen dargelegt.

Bevorzugte Ausführungsformen der Erfindung sind in den Zeichnungen dargestellt und werden nachfolgend beschrieben.

Fig. 1 zeigt schematisch den Ablauf des Reduzierungsverfahrens;

Fig. 2 stellt den Zusammenhang zwischen den Komponenten und den Zuständen der einzelnen Komponenten dar;

Fig. 3a, 3b zeigen eine Zusammenstellung verschiedener Komponenten mit den zugeordneten diskreten elektrischen

Zustandswerten sowie den möglichen Betriebs- und Komponentenzuständen;

Fig. 4 stellt die Beziehung zwischen den Grundbausteinen und den Komponenten dar;

Fig. 5 zeigt die Entstehung eines Reduktionsgraphen;

Fig. 6 zeigt das Prinzip der Entstehung eines reduzierten Teilgraphen aus einem beliebigen Teilgraphen;

Fig. 7 stellt die Arbeit von Bindungsoperatoren dar;

Fig. 8 zeigt einen Auszug aus einer Relationstabelle;

Fig. 9a-9d zeigen verschiedene Reduktionsverfahren:

Fig. 9a eine 0-Widerstandsreduktion,

Fig. 9b eine Schleifenreduktion,

Fig. 9c eine Serienreduktion,

Fig. 9d eine Parallelreduktion;

Fig. 10 zeigt eine Stern-Vieleck-Reduktion, bei der ein n-strahliger Widerstandsstern in ein vollständiges n-Eck überführt wird;

Fig. 11a stellt eine Spannungsquellenreduktion,

Fig. 11b eine Stromquellenreduktion und

Fig. 11c eine Äquivalenzoperation für Spannungsquellen mit Widerständen dar.

In Fig. 1 ist der Ablauf des Verfahrens zur Reduzierung der Anzahl der Simulationsschritte dargestellt. Außer den unbedingt erforderlichen Verfahrensschritten sind die zusätzlichen Schritte Aufteilen der Wissensdaten in Wissensdatentypen sowie Ermitteln und Zusammenfassen von räumlich begrenzt Einfluß nehmenden Grundbausteinen enthalten, die jeweils eine weitere vorteilhafte Ausgestaltung des erfindungsgemäßen Verfahrens darstellen.

Die wesentlichen Komponenten, die in der Peripherie von Automatisierungsgeräten auftreten, sind, wie in der Tabelle der Fig. 3 zusammengestellt, Widerstände, Schalter, Leitungen, Stecker, Lampen, Leitungsbündel, Sensoren, Relais, Sicherungen, Magnetventile, Motoren und Signalgeber. Ein Teil der Komponenten läßt sich in Aggregate A zusammenfassen. Die Aggregatinformationen mit den Strukturbeziehungen stehen üblicherweise in einer datenverarbeitungstechnischen Form zur Verfügung. Diese Informationen sind bereits während den Produktentwicklungsphasen z. B. in CAD-Systemen vorhanden oder können aus modellbasierten Programmen abgeleitet werden.

Ein Aggregat  $A_i$  ist eine Menge von Komponenten  $C_j$  mit einer lokalen Anordnung. Ein Aggregat beschreibt eine kleinste tauschbare Einheit. Die einzelnen Aggregate  $A_i$  sind durch die Pins  $PI_i$  der Komponenten  $C_j$  miteinander verbunden.

In der linken Bildhälfte von Fig. 2 sind die Kardinalitäten der Beziehungen von Aggregaten dargestellt. Für die allgemeine und systemunabhängige Darstellung von Komponenten können die Beziehungen wie folgt definiert werden:

- Ein Aggregat  $A_i$  besteht aus null oder mehreren Komponenten  $C_j$ . Eine Komponente  $C_j$  gehört genau zu einem Aggregat  $A_i$ .
- Eine elektrische Komponente  $C_j$  hat null oder mehrere Ports  $PO_k$ . Ein Port  $PO_k$  ist Bestandteil einer Komponente  $C_j$ .
- Jeder Port  $PO_k$  kann keinen oder mehrere Pins  $PI_m$  enthalten. Jeder Pin  $PI_m$  ist genau einem Port  $PO_k$  zugeordnet.
- Eine Verbindung  $CN_i$  bezieht sich auf zwei Pins  $PI_m$ . Ein Pin  $PI_m$  kann mit keiner oder einer Verbindung  $CN_i$  in Relation stehen.

Allen hier betrachteten Komponenten gemeinsam ist das in bestimmten Arbeitspunkten lineare Verhalten. Nichtlineare Komponenten wie z. B. Dioden werden durch prozessorgesteuerte Automatisierungsgeräte ersetzt. Zum einen bilden die Softwarefunktionen das nichtlineare Verhalten nach und zum anderen werden die nichtlinearen Komponenten in die Automatisierungsgeräte integriert, so daß das Verhalten im wesentlichen außerhalb der Systeme im Rahmen der relevanten Informationen linear beschreibbar ist.

Die mit diesen Komponenten verbundenen Fehler beschränken sich vorwiegend auf diskrete Fehler. Erfahrungsgemäß überwiegt in der Peripherie der Automatisierungsgeräte der Anteil der diskreten oder diskretisierbaren Fehler mit statischem Verhalten. Kontinuierliche Fehler können zwar nicht ausgeschlossen werden, sind aber im allgemeinen zu vernachlässigen.

Für die Simulation ist es ausreichend, die statischen Eigenschaften der Komponenten zu berücksichtigen. Dynamische Eigenschaften der hier betrachteten Komponenten spielen eine untergeordnete Rolle, da die dynamischen Verhaltenseigenschaften der betrachteten Komponenten keinen wesentlich höheren Informationsgewinn liefern.

Die nachfolgenden Betrachtungen konzentrieren sich aufgrund der praktischen Relevanz vorwiegend auf die passive Peripherie. Aktive Komponenten sind mit der gleichen Methode zu analysieren, spielen jedoch im allgemeinen eine untergeordnete Rolle. Bei einer rein passiven Peripherie zeigen alle Komponenten einen elektrischen Widerstand mit unterschiedlichen Widerstandswerten. In aktiven Schaltungen kommen zu diesen Widerständen Strom- und Spannungsquellen hinzu. Diese primären Komponenten werden als Grundbausteine oder atomare Einheiten bezeichnet.

Jeder Grundbaustein  $U_i$  verweist auf eine Komponente  $C_i$ . Die Grundbausteine oder atomaren Einheiten unterscheiden sich je nach Typ in aktive und passive Grundbausteine. Die aktiven Grundbausteine gliedern sich in Stromquellen  $QI$  und Spannungsquellen  $QV$ , hingegen enthalten die passiven Grundbausteine einfache Widerstände  $R_k$ . Der Typ des Grundbausteins ist durch  $type(U_i)$  charakterisiert. Der Grundbaustein  $U_i$  kann je nach Zustand der Komponente unterschiedliche diskrete Werte  $V_k$  annehmen. Die einzelnen diskreten Werte  $V_k$  der Komponente  $C_i$  sind durch die Zustände der Komponenten über eine Verhaltensbeschreibung  $B_k$  miteinander gekoppelt.

Bei den weiteren Überlegungen werden vor allem die passiven Grundbausteine betrachtet. Die aktiven Grundbausteine sind zur Vervollständigung aufgeführt.

Das Verhalten der Komponenten ist durch diskrete Zustände charakterisiert. Wie in Fig. 4 dargestellt ergeben sich hieraus für die Grundbausteine diskrete Werte, die unterschiedliche elektrische Zustände der Komponenten repräsentieren. Die einzelnen diskreten Zustände der Grundbausteine stehen in einer bestimmten Relation zu den Komponentenzu-

ständen, die das Normalverhalten und das Fehlverhalten einer Komponente umfassen. Darüber hinaus kann es für viele Anwendungen, wie z. B. für Simulationsverfahren zur Erzeugung von Diagnosewissen, von Interesse sein, in welchem Betriebszustand die Komponente sich befindet. Je nach Art und Weise welche physikalische Größe den Betriebszustandswechsel verursacht, sind elektrische und nicht-elektrische Betriebszustände zu unterscheiden. Die elektrischen Betriebszustände werden durch elektrische Größen der zu untersuchenden Grundbausteine direkt beeinflusst.

Jede Komponente  $C_i$  kann unterschiedliche Betriebszustände  $O_i$  annehmen. Die Betriebszustandsübergänge sind je nach Einflußfaktor von elektrischen und/oder nicht elektrischen Ereignissen und Größen abhängig. Sind mehrere Betriebszustände vorhanden, erfolgt der Zustandsübergang primär entweder durch ein elektrisches oder nicht-elektrisches Ergebnis. Die nicht-elektrischen Betriebszustände  $ONE_j$  können z. B. durch mechanische Einwirkungen erreicht werden.

Eine Komponente  $C_i$  kann elektrische Betriebszustände  $OE_j$  annehmen, wenn eine elektrische Wirkung einen Betriebszustandswechsel hervorruft. Einen Wechsel zwischen den Betriebszuständen ist ohne Einschränkung beliebig oft durchführbar.

Jede Komponente  $C_i$  besitzt mindestens zwei Komponentenzustände  $F_m$ , die weder von elektrischen Größen noch von äußeren Einwirkungen im Normalbetrieb verändert wird. Hinter den Komponentenzuständen  $F_m$  verbergen sich mindestens ein Fehlerzustand und der Normalzustand. Für Simulationsverfahren zur Erzeugung von Diagnosewissen ist es deshalb von primärer Bedeutung, den Komponentenzustand zu erkennen. Ein Komponentenzustandswechsel vom Normalzustand zu einem Fehlerzustand erfolgt nur einmal. Ausnahmen bilden die sporadischen Fehler.

Eine Glühlampe z. B. kann unabhängig von der Betriebsspannung defekt oder nicht defekt sein. Dieses der Komponente fest zugeordnete Verhalten ist der Komponentenzustand. Im Komponentenzustand nicht defekt sind je nach vorhandenen elektrischen Größen unterschiedliche Betriebszustände möglich. Im allgemeinen ist eine Unterscheidung zwischen aktivem Betrieb (die Lampe leuchtet) und passivem Betrieb (die Lampe leuchtet nicht) ausreichend.

Die einzelnen Komponenten können, je nach örtlicher Lage, dem vollständigen Gesamtsystem oder der Peripherie des Gesamtsystems zugewiesen werden. Prinzipiell kann das vollständige Gesamtsystem in das Modell miteinfließen, jedoch sind die interessierenden Informationen überwiegend in der Peripherie zu finden.

Die Peripherie des Gesamtsystems ist als Graph modellierbar, in dem die Grundbausteine die Kanten repräsentieren. Zur Verbindung der Grundbausteine sind Verbindungspunkte notwendig, die im Graph als Knoten auftreten.

Ein Graph  $G$  ist gegeben durch eine Menge an Knoten  $N$  und Kanten  $E$ . Jeder Knoten  $N_i$  ist die Verbindungsstelle von zwei oder mehreren Kanten  $E_j$ . Eine Kante  $E_i$  verbindet zwei Knoten  $N_i$  und  $N_j$ . Der Graph  $G$  heißt ungerichtet, wenn keine Knotenreihenfolge einer Kante existiert. Zu den Graphenelementen  $GE$  gehören sowohl die Kanten  $E_j$  als auch die Knoten  $N_i$ .

Der Grad oder  $\text{degree}(N_i)$  des Knotens ist durch die Anzahl der Kanten  $E_j$  an diesem Knoten bestimmt. Die mit dem Knoten  $N_i$  verbundenen Kanten  $E_j$  sind durch die  $\text{edges}(N_i)$  charakterisiert.

Ist eine Kante eine Schleife, d. h., sind die beiden Knoten der Kante  $N$  identisch, so liefert  $\text{loop}(N_i)$  ein TRUE, andernfalls ist das Resultat FALSE. Das Ergebnis der Funktion  $\text{chain}(N_i, N_j)$  ist eine Menge von Kanten  $E_i$ , die unmittelbar in einer Kette mit den Elementen  $N_i$  und  $N_j$  liegen. Jeder innere Knoten einer nicht geschlossenen Kette hat den Knotengrad 2, die äußeren Knoten besitzen den Knotengrad 3 oder größer.

Für die Peripherie in der elementaren Grundebe, d. h. in der tiefsten Modellierungsebene, weist ein Graph die nachfolgenden Eigenschaften auf:

Die Peripherie kann als ungerichteter Graph  $G_{ep} = (N_j, Z_i)$  beschrieben werden. Der Graph ist durch die Kanten  $E_i$  mit den atomaren Einheiten  $U_i$  und den Knoten  $N_i$  gekennzeichnet. Ein Knoten  $N_i$  ist die Verbindungsstelle von zwei oder mehreren Grundbausteinen  $U_i$ . Das gesamte Verhalten der elektrischen Peripherie charakterisiert der elementare Peripheriegraph  $G_{ep}$ .

In einem elementaren Peripheriegraph  $G_{ep}$  können die Knotenpotentiale  $P_j$  oder die Ströme  $I_i$  durch einen Grundbaustein meßbar sein. Die Meßbarkeit des Knotenpotentials  $P_i$  oder des Zweigstromes  $I_i$  hängen von der örtlichen Lage und der Struktur des gesamten Systems ab und können die Werte meßbar oder nicht meßbar annehmen. Das Potential  $P_i$  am Knoten  $N_i$  enthält darüber hinaus einen festen Wert  $V_B$  oder einen unbekannten Wert  $V_{vi}$ . Für die Zweigströme  $I_i$  ergeben sich analog feste oder unbekannte Werte. Die Meßbarkeit einer physikalischen Größe ist eindeutig einem Grundbaustein  $U_i$  oder einem Knoten  $N_i$  zuweisbar. Die Peripherie läßt sich somit als ungerichteter Graph darstellen.

Zwischen den aus beispielsweise CAD-Daten entnehmbaren Informationen, die die Struktur und die einzelnen Komponenten beschreiben und den Zuständen der einzelnen Komponenten ergeben sich die in Fig. 2 aufgezeigten Relationen.

- Jeder Komponente  $C_i$  kann mindestens ein Komponentenzustand  $F_m$  zugeordnet werden. Jeder Komponentenzustand  $F_m$  gehört genau zu einer Komponente  $C_i$ .

- Der Betriebszustand  $O_i$  ist genau einer Komponente  $C_i$  zuzuordnen. Eine Komponente  $C_i$  kann keinen oder mehrere Betriebszustände  $O_i$  enthalten.

- Der Betriebszustand  $O_i$  kann durch meßbare Größen eindeutig verursacht werden.

- Zu jeder Komponente  $C_i$  gehört mindestens ein Knoten.

- Jeder Komponentenzustand  $F_m$  ist eindeutig durch die diskreten Werte der Grundbausteine oder atomaren Einheiten beschrieben.

Auf der Basis des oben eingeführten elementaren Peripheriegraphen  $G_{ep}$  läßt sich eine Graphenreduktion durchführen, in der die wesentlichen Verhaltenseigenschaften hinsichtlich der beobachtbaren physikalischen Größen enthalten sind. Wie in Fig. 5 dargestellt, kann ein Reduktionsgraph  $G_r$  aus mehreren Untergraphen bestehen. Zu diesen gehören der elementare Peripheriegraph  $G_{ep}$ , mehrere Zwischengraphen  $G_i$  und ein Kopfgraph  $G_b$ . Die Knoten und Kanten der einzelnen Untergraphen sind miteinander verbunden und bilden die Reduktionsvorschrift ab.

Zum Aufbau des Reduktionsgraphen finden die aus der Netzwerkanalyse bekannten Grundoperationen Anwendung. Die Grundüberlegung ist, daß unter bestimmten Voraussetzungen Zweige in einem elektrischen Netzwerk durch einen

gleichwertigen Zweig einfacherer Struktur darstellbar sind. Neben den einzelnen Reduktionsoperationen benötigt der effiziente Aufbau eines Reduktionsgraphen Äquivalenzoperationen, in denen die Netzwerkstruktur in eine äquivalente Darstellungsform umgewandelt wird und diese schließlich den Ausgangspunkt für einen weiteren Reduktionsschritt bildet. Die Äquivalenzoperationen führen keine Reduktion der Netzwerkstruktur durch, d. h., die Anzahl der Komponenten im übergeordneten Graphen verglichen mit der des aktuellen Graphen verändert sich nicht.

Die hierzu notwendigen Operationen sind in Fig. 9a–9d im einzelnen dargestellt. Zu beachten ist, daß hinter den einzelnen Werten der Grundbausteine oder atomaren Einheiten nicht nur ein einzelner Wert, sondern eine Menge möglicher Werte steht. Diese Werte beschreiben das Verhalten der Komponente.

Eine Serienreduktion wie in Fig. 9c dargestellt ist anwendbar auf Grundbausteine des Typs Widerstände  $R_k$  und Spannungsquellen  $Q_v$ . Für diese Grundbausteine  $U_i$  mit den Werten  $V_k$  gilt:

$$V_{se} = \sum_{i=1}^{n-1} V_{i(i+1)} \quad (0.1)$$

für Widerstände  $V = R$

für Spannungsquellen  $V = U$

Für die diskreten Werte  $V = \{0, \infty\}$  gilt:

$$\exists V_{i(i+1)} = \infty \Rightarrow V_{se} = \infty \quad (0.2)$$

$$\forall V_{i(i+1)} = 0 \Rightarrow V_{se} = 0 \quad (0.3)$$

Für den Typ des Grundbausteins Widerstände  $R_k$  und Stromquellen  $Q_i$  läßt sich das in Fig. 9d gezeigte Parallel-Reduktionsverfahren einsetzen:

$$V_{pa} = \sum_{i=1}^n V_i \quad (0.4)$$

für Widerstände  $R = 1/V$

für Stromquellen  $QI = V$

Für die diskreten Werte  $V = \{0, \infty\}$  gilt:

$$\forall V_{i(i+1)} = 0 \Rightarrow V_{pa} = 0 \quad (0.5)$$

$$\exists V_{i(i+1)} = \infty \Rightarrow V_{pa} = \infty \quad (0.6)$$

Die Anwendbarkeit der in Fig. 10 gezeigten Stern-Vieleck-Reduktion beschränkt sich auf Widerstände  $R_k$ . Allgemein gilt für die Berechnung des Eck-Widerstandswertes:

$$R_{\nu\mu} = R_{\nu 0} R_{\mu 0} \sum_{m=1}^n \frac{1}{R_{m0}} \quad (0.7)$$

Mit den Schenkelwiderständen:

$$R_{sm} = \{R_{m0} | m = \mu \vee m = \nu\} \quad (0.7)$$

und den Restwiderständen:

$$R_{RRm} = \{R_{m0} | 0 \leq m \leq n \setminus \{\mu, \nu\}\} \quad (0.9)$$

gelten für die diskreten Werte  $R = \{0, \infty\}$  nach einer Grenzwertbetrachtung nachfolgende Relationen:

$$\exists R_{Sm} = \infty \Rightarrow R_{\mu\nu} = \infty \quad (0.10)$$

$$\forall R_{Sm} = 0 \Rightarrow R_{\mu\nu} = 0 \quad (0.11)$$

$$\forall R_{Sm} = + \wedge \exists R_{RRm} = 0 \Rightarrow R_{\mu\nu} = \infty \quad (0.12)$$

$$\forall R_{Sm} = + \wedge \forall R_{RRm} \neq 0 \Rightarrow R_{\nu\mu} = R_{\nu 0} R_{\mu 0} \sum_{m=1}^n \frac{1}{R_{m0}} \quad (0.13)$$

$$\exists R_{Sm} = 0 \wedge \exists R_{Sm} = + \wedge \forall R_{RRm} \neq 0 \Rightarrow R_{\mu\nu} = \sum R_{Sm} \quad (0.14)$$

$$\exists R_{Sm} = 0 \wedge \exists R_{Sm} = + \wedge \exists R_{RRm} = 0 \Rightarrow R_{\mu\nu} = k \sum R_{Sm} \quad (0.15)$$

mit  $k$  = Anzahl der 0-wertigen Restwiderstände

Besteht ein Stern aus  $n$  Kanten und  $u$  Knoten, so umfaßt der reduzierte Teilgraph  $n(n-1)/2$  Kanten und  $u-1$  Knoten. Dies führt zu einem Reduktionsaufwand für das Hinzufügen und Entfernen von Kanten in der Größenordnung  $O(n^2)$ .

Die Stern-Vieleck-Reduktion kann nach einer sortierten Liste durchgeführt werden, in der zuerst die Knoten höherer Ordnung bearbeitet werden. Die praktischen Auswirkungen auf den reduzierten Graphen sind jedoch nicht relevant, da die Reihenfolge auf die Stern-Vieleck-Reduktion nur die Form des Reduktionsgraphen ändert, nicht aber den Kopfgraphen.

Die in Fig. 11a dargestellte Spannungsquellenreduktion gilt in dieser Anordnung für Widerstände  $R_k$  und Spannungsquellen  $QV_1$  mit dem Knotengrad  $n > 2$ .

Für eine Kombination aus Widerständen  $R_k$  und Stromquellen  $QI_1$  gilt die in Fig. 11b dargestellte Stromquellenreduktionsoperation. Die Stromquellenreduktion gilt auch für Knoten höherer Ordnung der gleichen Struktur.

Die in Fig. 11c gezeigte Äquivalenzoperationen für Spannungsquellen mit Widerstand hat zum Ziel, die einzelnen Grundbausteine nach ihrem Typ zu sortieren und für eine anschließende Serienreduktion vorzubereiten.

Die Werte der einzelnen Graphenelemente GE repräsentieren Betriebs- und Komponentenzustände. Das primäre Ziel ist es, die Komponentenzustände und hieraus den Systemzustand zu ermitteln. Die Menge aller Systemzustände ist somit die Permutation aller einzelnen Komponentenzustände, die wiederum durch die diskreten Werte dargestellt sind. Der Gesamt- oder Teilsystemzustand setzt sich aus den Permutationen der diskreten Werte  $V_i$  der einzelnen Grundbausteine  $U_i$  zusammen. Um das gesamte Verhalten eines untergeordneten Graphen in den übergeordneten Graphen zu übertragen, ist die Erstellung der Permutationen zwischen den einzelnen Werten notwendig.

Die Operationen 0-Widerstandsreduktion, Schleifenreduktion, Spannungsquellenreduktion und Stromquellenreduktion entfernen Graphenelemente, ohne neue Elemente hinzuzufügen. Für diese Operationen sind keine besonderen Verfahren notwendig. Die Operationen Serien-, Parallel- und die Stern-Vieleck-Reduktion fügen neue Werte in den Graphen ein, so daß die Erzeugung der Permutationen und die entsprechende Verarbeitung notwendig wird.

Fig. 6 zeigt die prinzipielle Verarbeitung der einzelnen Werte zur Erzeugung eines reduzierten Teilgraphen  $G_{\text{sub red}}$  aus einem beliebigen Teilgraphen  $G_{\text{sub org}}$ . Zur Berechnung der Permutationen wird ein sequentielles Permutierungsverfahren verwendet. Jedoch ist für nachfolgende Ableitungsverfahren wesentlich, daß die Reihenfolge der einzelnen Werte bestehen bleibt und eine lexikographische Ordnung nicht durchgeführt wird.

Die Anzahl der Werte des Ergebnisses  $z$  einer mathematischen Funktion der Form

$$f(x_1, x_2, \dots, x_u) = z \quad (0.16)$$

mit  $x_i$  als Parameter und den diskreten Werten

$$x_i \in \{x_{i1}, x_{i2}, \dots, x_{iu}\} \quad (0.17)$$

ergibt sich aus

$$v_z = \prod_{i=1}^u v_i \leq \max_{i=1}^u (v_i)^u \quad (0.18)$$

Mit den Gleichungen (0.2), (0.5) und (0.10) reduziert sich die Anzahl der diskreten Ergebniswerte nach den Reduktionsoperationen Serien-, Parallel- und Stern-Vieleck-Reduktion. Verallgemeinert gilt:

$$v_{\text{redu oper}} \leq \prod_{i=1}^u v_i \leq \max_{i=1}^u (v_i)^u \quad (0.19)$$

Wie aus Fig. 3 zu entnehmen ist, treten in den elektrischen Komponenten vorwiegend die diskreten Werte 0 und  $\infty$  auf, so daß für praktische Anordnungen gilt:



$$v_{\text{redu oper}} \ll \prod_{i=1}^n v_i \leq \max_{i=1}^n (v_i)^n \quad (0.20)$$

Jede Stern-Vieleck-Reduktion mit einem Knotengrad größer als 3 führt dazu, daß die Anzahl der Kanten im Graphen zunimmt. Die Stern-Vieleck-Reduktion ergänzt die ursprünglichen Komponentenzustände durch weitere Kombinationen, die nachfolgend als Pseudozustände bezeichnet werden. Der Zuwachs von  $n(n-1)/2$  Kanten führt bei dieser Reduktion zu einem exponentiellen Wachstum der Pseudozustände. Dies kann einerseits zu einer rapiden Erhöhung des Rechen- bzw. Speicheraufwands von einem Reduktionsschritt zum nächsten führen. Bei einer Verarbeitung, wie in Fig. 5 dargestellt ist, steigt bei jedem Stern-Vieleck-Reduktionsschritt die Anzahl der Zustände im Gesamtsystem an.

Um die Permutationen von nicht realistischen Wertekombinationen, verursacht durch die Stern-Vieleck-Reduktion, einzuschränken, wird ein sogenannter Bindungsoperator eingeführt. Der Bindungsoperator kennzeichnet alle permutierten Werte als sogenannten Permutationsblock. Die Berechnung der Werte in einem Reduktionsschritt erfolgt schließlich nicht durch die einzelnen Werte, sondern bei der Berechnung wird der gesamte Permutationsblock mit allen Bindungen berücksichtigt.

Bei der Erzeugung der Permutationen müssen auch Elemente berücksichtigt werden, die nicht zur Berechnung notwendig sind, aber durch den Bindungsoperator zusammenhängen. Fig. 7 stellt die Verarbeitung der Bindungsoperationen dar. Trotz Zunahme der Elemente in einer Stern-Vieleck-Reduktion verhindert der Bindungsoperator Pseudokomponentenzustände.

Dieses Verfahren zur Berechnung der einzelnen diskreten Widerstandswerte ermöglicht, beliebige Kombinationen während des Reduktionsvorgangs zu berücksichtigen und andere nicht erwünschte Kombinationen von vorne herein auszuschließen. So können beispielsweise bestimmte Wissensdatentypen wie z. B. Einfachfehler spezifiziert und durch dieses Verfahren reduziert werden.

Durch das Abstrahieren und die Reduktion des peripheren Widerstandsgraphen kann die ein oder andere Verhaltenseigenschaft, dargestellt durch die einzelnen diskreten Werte, mehrfach auftreten. Es ist jedoch ausreichend, jeweils nur eine Verhaltenseigenschaft zu berücksichtigen.

Werte, die keine Bindung besitzen, lassen sich auf das einmalige Auftreten reduzieren. Bei Elementen mit einer Bindung ist eine Kombination für die weitere Widerstandsdarstellung ausreichend. Die Weiterreduktion kann direkt nach einer Reduktionsoperation erfolgen, sie kann aber auch erst nach mehreren Reduktionsschritten stattfinden.

Ausgangspunkt für das Reduktionsverfahren ist ein ungerichteter elementarer Grundgraph  $G_{ep} = (U, N)$  mit Knoten  $N_i$  und den atomaren Einheiten  $U_i$  als Kanten. Die Reduktion des Netzwerks erfolgt mit den beschriebenen Grundoperationen. Nach jedem Reduktionsdurchlauf verringert sich die Anzahl der Knoten  $N_i$ . Für die Anzahl der atomaren Einheiten  $U_i$  im reduzierten Graphen ist eine Verkleinerung, verursacht durch die Stern-Vieleck-Reduktion, nicht sichergestellt. Der elementare Grundgraph  $G_{ep}$  wird so lange bearbeitet, bis kein Knoten und keine Kante mehr aus den Graphen entfernt werden können.

Am Ende des Reduktionsverfahrens besteht der Kopfgraph aus Knoten mit festem oder meßbaren Potentialen. Alle Knoten mit unbekannten Potentialen sind im Kopfgraphen  $G_h$  entfernt. Atomare Einheiten  $U_i$ , die eine Strommessung enthalten, verbleiben im reduzierten Graphen.

Für einen elementaren Grundgraphen  $G_{ep} = (U, N)$  mit  $n$  Knoten  $N_i$ , und  $n$  atomaren Einheiten  $U_i$  beträgt die Zeitkomplexität des Algorithmus  $O(n^2)$ . Für komplexe Gesamtsysteme mit einer Vielzahl von Komponenten ist die für die Erstellung der Reduktionsgraphen notwendige Zeit beträchtlich. Um diesem Verhalten zu begegnen ist es vorteilhaft, vor dem eigentlichen Reduktionsverfahren eine Clusterung durchzuführen, in der die Lokalität der einzelnen Komponenten ermittelt wird.

Die grundlegende Überlegung, die sich hinter der Clusterung verbirgt, ist, daß sich nicht alle Betriebs- und Fehlzustände von Komponenten auf alle Systemgrößen auswirken. Es existieren somit räumlich unabhängige Bereiche, in denen sich gewisse Zustände von Komponenten auswirken, und Bereiche, die nicht von allen Komponenten beeinflusst werden. Um den Ressourcenbedarf für die Simulation zu minimieren ist es zweckmäßig, solche räumlichen Bereiche mit deren Elementen zu identifizieren. Die Clusterung ermöglicht es, ein systemumfassendes Problem auf die wesentlichen Elemente im System zu reduzieren und somit eine Eingrenzung des globalen Problems auf ein begrenztes lokales Gebiet durchzuführen.

Ein Cluster  $C$  besteht aus einer bestimmten Anzahl von Elementen. Alle Zustände der im Cluster liegenden Elemente wirken sich nur auf Größen und Fehlermeldungen in diesem lokalen Clusterbereich aus.

Die einzelnen Bereiche können sich auf unterschiedliche Abstraktionsebenen beziehen. Eine Abstraktionsebene bildet der elementare Peripheriegraph. Durch die gegenseitige Kopplung der einzelnen Grundbausteine oder atomaren Einheiten über die entsprechenden Komponenten existiert eine zweite Ebene, die Clusterbereiche auf Komponentenebene beinhaltet.

Eine Clusterung auf der Ebene des elementaren Peripheriegraphen umfaßt als Elemente die Grundbausteine. Im Gesamtsystem gehört jede mögliche Beobachtung einem Cluster an.

Ein Grundcluster oder atomarer Cluster  $C_a$  ist ein zusammenhängender Pfad von Grundbausteinen oder atomaren Einheiten. Jeder Grundbaustein  $U_i$  und die dazwischen liegenden Knoten  $N_i$  können genau einem Grundcluster  $C_a$  zugeordnet werden. Alle Werte, die die atomaren Einheiten und die Knoten annehmen, wirken sich nur auf die Größen im Grundcluster  $C_a$  aus.

Mit jedem beliebigen Graphen-Transversierungsverfahren lassen sich die zusammenhängenden Komponenten erkennen. In verschiedenen Literaturstellen sind Algorithmen zur Transversierung von Graphen zu finden. Gegenüber den gewöhnlichen Transversierungsverfahren muß jeder durchlaufene Knoten und jede durchlaufene Kante in eine Liste eingetragen und als bearbeitet markiert werden.

Das Ergebnis der Clusterung auf atomarer Ebene ist eine Liste mit Clustern, die jeweils Verweise auf die einzelnen

Grundbausteine und die Nachbarknoten, die zu diesen Clustern gehören, enthalten. Ein komplexer elementarer Peripheriegraph zerfällt durch die Clusterung in räumlich begrenzte Wissensdatenbereiche. Die in einem Cluster vorhandenen Graphenelemente besitzen auf atomarer Ebene keine Beziehungen zu Graphenelementen eines anderen Clusters. Jeder Cluster bildet somit ein abgeschlossenes Teilproblem und ist somit auch als Fokussierungsbereich zu bezeichnen.

Verwendet die Erstellung des Reduktionsgraphen nicht den gesamten elementaren Peripheriegraph, sondern werden die einzelnen Cluster berücksichtigt, so ist lokal betrachtet nach wie vor die Zeitkomplexität mit  $2n$  Graphenelementen in einem Cluster  $O(n^2)$ . Aus der globalen Komplexitätsbetrachtung geht jedoch hervor, daß bei  $k$  Clustern der Aufruf des Reduktionsverfahrens nur eine Zeitkomplexität von  $O(k)$  besitzt. Für komplexe Systeme ist dies von wesentlicher praktischer Bedeutung, da hierdurch die realistische Bearbeitung von Systemen mit mehreren tausend Komponenten möglich ist.

Die Clusterung bietet darüber hinaus für das Gesamtsystem die Möglichkeit der parallelen Erstellung des Reduktionsgraphens. Im günstigsten Fall können alle Cluster parallel bearbeitet werden. Unter diesen Voraussetzungen stellt sich für die Erstellung des Reduktionsgraphens mit  $n'$  lokalen Graphenelementen die Zeitkomplexität von  $O(n'^2)$  ein.

Einzelne Komponenten können eine Beziehung zu mehreren Grundbausteinen besitzen. Die Zustände der Komponenten betreffen ebenfalls eine oder mehrere Grundbausteine. Hierdurch entstehen Relationen zwischen Grundbausteinen verschiedener Grundcluster.

Ein komponentenorientierter Cluster  $C_c$  umfaßt einen oder mehrere Grundcluster  $C_a$  mit dessen Grundbausteinen  $U_i$  und Knoten  $N_j$ . Die Grundcluster  $C_a$  sind durch die Relationen zwischen den Komponenten und den Grundbausteinen miteinander verbunden. Berücksichtigt werden aber nur die Komponenten in der Peripherie.

Für den komponentenorientierten Cluster gelten die gleichen Erstellungsverfahren wie für die Grundcluster. Das Ergebnis der Clusterung auf Komponentenebene ist eine Liste mit Clustern, die jeweils Verweise auf die Grundcluster und Verweise auf Komponenten besitzt. Die komponentenorientierten Cluster bilden die Grundlage für die Relationen zwischen den Grundbausteinen im Kopfgraphen und den Zuständen der Komponenten.

Nach einer durchgeführten Simulation müssen die Simulationsergebnisse wieder den Komponentenzuständen zugeordnet werden. Die erforderlichen Informationen stellt eine Relationstabelle zur Verfügung. Für jeden komponentenorientierten Cluster beschreibt eine Relationstabelle alle möglichen Komponentenzustände des Clusters mit den entsprechenden diskreten Werten der Grundbausteine. Die Tabelle läßt sich auch auf ausgewählte Kombinationen der Komponentenzustände begrenzen. Während des gesamten Reduktionsverfahrens wird diese Tabelle ständig aktualisiert. Am Ende des Reduktionsverfahrens befinden sich in der Relationstabelle die zu Beginn spezifizierten Komponentenzustände mit den diskreten Werten des Kopfgraphen. Fig. 8 veranschaulicht einen Auszug aus einer Relationstabelle.

Mit Hilfe der Relationstabelle können bei einem bekannten diskreten Wert aus dem Kopfgraphen relativ einfach die dazugehörigen Komponentenzustände ermittelt werden. Wird hinter die letzte Spalte der Relationstabelle noch das Simulationsergebnis angefügt, so ergibt sich z. B. bei der Erzeugung von Diagnosewissen die klassische Fehlertabelle mit den Komponentenzuständen und den entsprechenden Systemgrößen. Die Größe der Tabelle kann auf bestimmte Kombinationen begrenzt werden.

Beispielhaft soll an einem Diagnosesystem für ein Gesamtfahrzeugsystem eine Abschätzung der notwendigen Simulationen ohne und mit dem Reduzierungsverfahren vorgenommen werden. Vollständigkeitshalber sind für die möglichen Fehlerannahmen Einfach- und Mehrfachfehler dargestellt. Zuerst findet eine Betrachtung der Fehlerzustände statt. Anschließend sind die Ansteuerungen diskutiert, da diese für die Fehlerauswirkung und Diagnose eine wichtige Rolle spielen.

	Komponenten		Fehlerannahmen	
	Anzahl	Fehlerzustände	einfach	mehrfach
Komponenten				
allgemeine Elektrik	13	2	26	$1.5 \cdot 10^6$
Leitungen	26	8	208	$6.4 \cdot 10^{24}$
Stecker	6	1	6	64
Knoten	2	1	2	4
Endhülsen	3	1	3	8
<b>Fehlerzustände je Steuergerät</b>			<b>245</b>	<b><math>2.1 \cdot 10^{34}</math></b>

Annahmen: Im Fahrzeug befinden sich 26 Steuergeräte.

	Fehlerannahmen	
	einfach	mehrfach
<b>Fehlerzustände eines Gesamtfahrzeugs</b>	<b>6370</b>	<b><math>2.1 \cdot 10^{892}</math></b>

Annahmen: Mindestens jeder zweite Einfachfehler hat die gleichen Auswirkungen und wird deshalb als redundant erkannt und entfernt.

	Fehlerannahmen	
	einfach	
<b>Fehlerzustände eines Gesamtfahrzeugs</b>	<b>3185</b>	

## Ansteuerungen

Ansteuerungen	Anzahl	Zustände	Kombinationen
je Steuergerät	6	2	64
<b>Gesamtfahrzeug</b>	<b>156</b>	<b>2</b>	<b>9 10<sup>46</sup></b>

Annahmen: Die Lokalität der Auswirkungen einer Ansteuerung wird durch die Hierarchisierung erkannt, so daß im Durchschnitt nur die Hälfte der möglichen Ansteuerungen je Steuergerät mit einer anderen gekoppelt sind

Ansteuerungen	Anzahl	Zustände	Kombinationen
<b>Gesamtfahrzeug</b>	<b>26</b>	<b>1/2x2<sup>6</sup></b>	<b>832</b>

Das Produkt der Fehlerzustände und der Ansteuerungsmöglichkeiten ergibt die Anzahl der notwendigen Simulationen.

Gesamtzahl der Simulationen	Fehlerzustände	Ansteuerungsmöglichkeiten	Simulationsfälle
ohne Reduktionsverfahren	6370	9 10 <sup>46</sup>	6 10 <sup>50</sup>
mit Reduktionsverfahren	3185	832	2.6 10 <sup>6</sup>

Zusätzlich zu den obigen Reduktionseigenschaften reduziert sich die Anzahl der Komponenten durch die Ersatzwertbildung. Aus verschiedenen Versuchen zeichnet sich ab:

Annahmen: Mindestens 50% aller Komponenten werden im Reduktionsverfahren entfernt. Bei einer quadratischen Relation zwischen Anzahl der Komponenten und Simulationszeit ergibt sich ein Reduktionsfaktor von 25%.

Unter der Annahme, daß ein Simulationsschritt eine Sekunde benötigt, stellen sich für die Simulationszeiten folgende Werte ein:

Simulationsdauer für die Gesamtfahrzeugelektrik bei 1s Rechenzeit je Simulation	
Gesamtfahrzeug ohne Reduktionsverfahren	6 10 <sup>50</sup> s
Gesamtfahrzeug mit Reduktionsverfahren	6.5 10 <sup>5</sup> s

Ein Vergleich der beiden Zahlen ergibt somit eine Reduktion des Simulationsaufwands mit dem oben beschriebenen Verfahren in Höhe von 10<sup>45</sup> für ein Gesamtfahrzeug.

## Patentansprüche

1. Verfahren zur Reduzierung der Anzahl der erforderlichen Simulationsschritte in einem Simulationsverfahren zur rechnergesteuerten Erzeugung von Wissensdaten über ein aus mehreren elektrisch ansteuerbaren Bauteilen bestehendes Gesamtsystem mit den Schritten

- Zerlegen der elektrisch ansteuerbaren Bauteile in Komponenten, die eine oder mehrere Grundbausteine umfassen
- Erfassen der elektrischen Verbindungen zwischen den Grundbausteinen und zwischen den Komponenten
- Zuordnen von diskreten elektrischen Zustandswerten zu den Grundbausteinen
- Definieren der Betriebszustände und der möglichen Komponentenzustände der einzelnen Komponenten in Relation zu den elektrischen Zustandswerten der zu einer Komponente gehörenden Grundbausteine
- Festlegen der für die Wissensdatenerzeugung notwendigen und an den Grundbausteinen und Komponenten meßbaren Größen
- Ermitteln von Grundbausteinen, die auf die festgelegten meßbaren Größen keinen Einfluß haben und
- Zusammenfassen und Eliminieren der ermittelten einflußlosen Grundbausteine.

2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, daß vor dem Schritt Festlegen der meßbaren Größen die zu erzeugenden Wissensdaten in Wissensdatentypen aufgeteilt werden und das Festlegen der an den Grundbausteinen und Komponenten meßbaren Größen nur für einen oder mehrere Wissensdatentypen erfolgt.

3. Verfahren nach Anspruch 1 oder 2, dadurch gekennzeichnet, daß nach dem Schritt Festlegen der meßbaren Größen jene Grundbausteine ermittelt und zusammengefaßt werden, die auf denselben räumlich begrenzten Wissensdatenbereich Einfluß nehmen, zur Bildung von lokalen Grundclustern.
4. Verfahren nach Anspruch 3, dadurch gekennzeichnet, daß Komponentencluster gebildet werden, die aus einem oder mehreren Grundclustern bestehen.
5. Verfahren nach einem der Ansprüche 1 bis 4, dadurch gekennzeichnet, daß die Wissensdaten durch Einbringen von Erfahrungswissen über die elektrisch ansteuerbaren Bauteile oder das Gesamtsystem reduziert werden auf die Wissensdaten, die die als wesentlich betrachteten Komponentenzustände betreffen.
6. Verfahren nach einem der Ansprüche 1 bis 5, dadurch gekennzeichnet, daß die Wissensdaten Diagnosewissen sind.
7. Verfahren nach einem der Ansprüche 1 bis 6, dadurch gekennzeichnet, daß die Komponenten Widerstände, Schalter, Leitungen, Lampen, Leitungsbündel, Stecker, Relais, Sicherungen, Magnetventile, Motoren, Signalgeber, Masseknoten und Spannungsquellen umfassen.
8. Verfahren nach einem der Ansprüche 1 bis 7, dadurch gekennzeichnet, daß die Grundbausteine Widerstände, Stromquellen und Spannungsquellen umfassen.
9. Verfahren nach einem der Ansprüche 1 bis 8, dadurch gekennzeichnet, daß es sich bei den diskreten elektrischen Zustandswerten um Widerstandswerte, Stromwerte oder Spannungswerte handelt.
10. Verfahren nach einem der Ansprüche 1 bis 9, dadurch gekennzeichnet, daß die Betriebszustände die Zustände offen oder geschlossen, Lampe ein oder aus und Relaiskontakt aktiv oder passiv umfassen.
11. Verfahren nach einem der Ansprüche 6 bis 10, dadurch gekennzeichnet, daß die Komponentenzustände die Zustände störungsfrei, defekt, geschlossen und kein Kontakt, offen und Kontakt, Unterbrechung, Kurzschluß und blockierter Rotor umfassen.
12. Rechner zur Durchführung eines Verfahrens zur Reduzierung der Anzahl der erforderlichen Simulationsschritte in einem Simulationsverfahren zur rechnergesteuerten Erzeugung von Wissensdaten über ein aus mehreren elektrisch ansteuerbaren Bauteilen bestehendes Gesamtsystem, enthaltend:
- Mittel zum Zerlegen der elektrisch ansteuerbaren Bauteile in Komponenten, die eine oder mehrere Grundbausteine umfassen
  - Mittel zum Erfassen der elektrischen Verbindungen zwischen den Grundbausteinen und zwischen den Komponenten
  - Mittel zum Zuordnen von diskreten elektrischen Zustandswerten zu den Grundbausteinen
  - Mittel zum Definieren der Betriebszustände und der möglichen Komponentenzustände der einzelnen Komponenten in Relation zu den elektrischen Zustandswerten der zu einer Komponente gehörenden Grundbausteine
  - Mittel zum Festlegen der für die Wissensdatenerzeugung notwendigen und an den Grundbausteinen und Komponenten meßbaren Größen
  - Mittel zum Ermitteln von Grundbausteinen, die auf die festgelegten meßbaren Größen keinen Einfluß haben und
  - Mittel zum Zusammenfassen und Eliminieren der ermittelten einflußlosen Grundbausteine.
13. Rechner nach Anspruch 12 zusätzlich enthaltend Mittel zum Aufteilen der Wissensdaten in Wissensdatentypen und zum Festlegen der an den Grundbausteinen und Komponenten meßbaren Größen nur für einen oder mehrere Wissensdatentypen.
14. Rechner nach Anspruch 12 oder 13 zusätzlich enthaltend Mittel zum Ermitteln und Zusammenfassen jener Grundbausteine, die auf denselben räumlich begrenzten Wissensdatenbereich Einfluß nehmen, zur Bildung von lokalen Grundclustern.
15. Rechner nach Anspruch 14 zusätzlich enthaltend Mittel zur Bildung von Komponentenclustern, die aus einem oder mehreren Grundclustern bestehen.
16. Rechner nach einem der Ansprüche 12 bis 15 zusätzlich enthaltend Mittel zum Einbringen von Erfahrungswissen über die elektrisch ansteuerbaren Bauteile oder das Gesamtsystem und Mittel zum Reduzieren der Wissensdaten auf die Wissensdaten, die die als wesentlich betrachteten Komponentenzustände betreffen.
17. Rechner nach einem der Ansprüche 12 bis 16, dadurch gekennzeichnet, daß die Wissensdaten Diagnosewissen sind.
18. Rechner nach einem der Ansprüche 12 bis 17, dadurch gekennzeichnet, daß die Komponenten Widerstände, Schalter, Leitungen, Lampen, Leitungsbündel, Stecker, Relais, Sicherungen, Magnetventile, Motoren, Signalgeber, Masseknoten und Spannungsquellen umfassen.
19. Rechner nach einem der Ansprüche 12 bis 18, dadurch gekennzeichnet, daß die Grundbausteine Widerstände, Stromquellen und Spannungsquellen umfassen.
20. Rechner nach einem der Ansprüche 12 bis 19, dadurch gekennzeichnet, daß es sich bei den diskreten elektrischen Zustandswerten um Widerstandswerte, Stromwerte oder Spannungswerte handelt.
21. Rechner nach einem der Ansprüche 12 bis 20, dadurch gekennzeichnet, daß die Betriebszustände die Zustände offen oder geschlossen, Lampe ein oder aus und Relaiskontakt aktiv oder passiv umfassen.
22. Rechner nach einem der Ansprüche 17 bis 21, dadurch gekennzeichnet, daß die Komponentenzustände die Zustände störungsfrei, defekt, geschlossen und kein Kontakt, offen und Kontakt, Unterbrechung, Kurzschluß und blockierter Rotor umfassen.

---

Hierzu 12 Seite(n) Zeichnungen

---

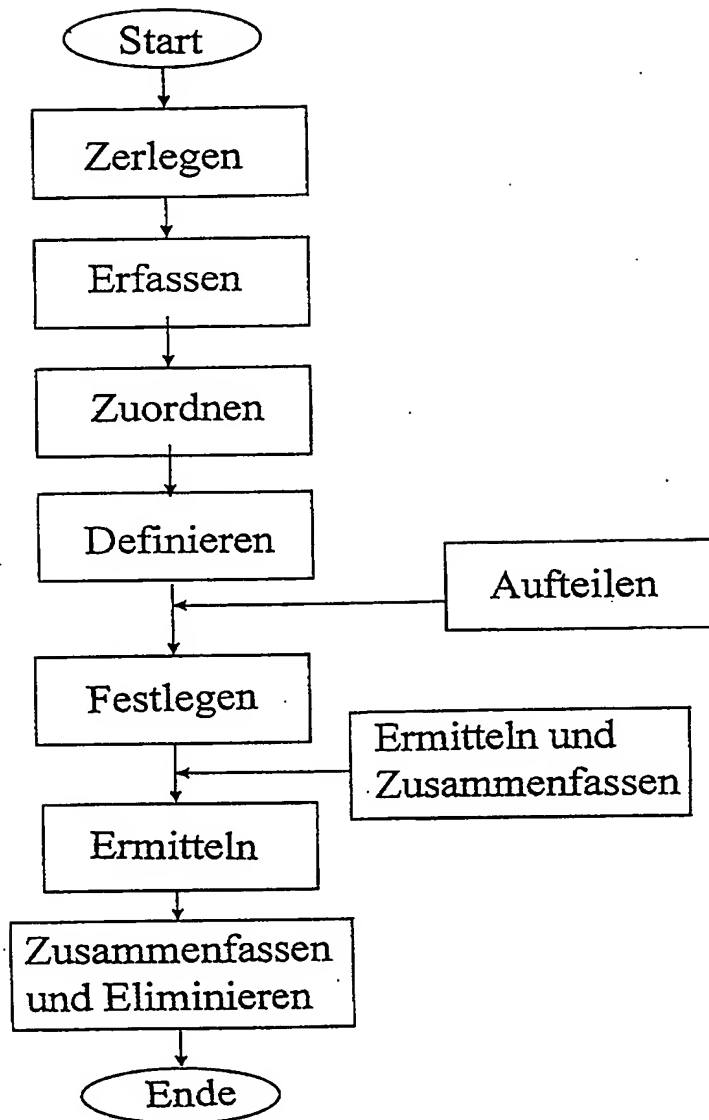


Fig. 1

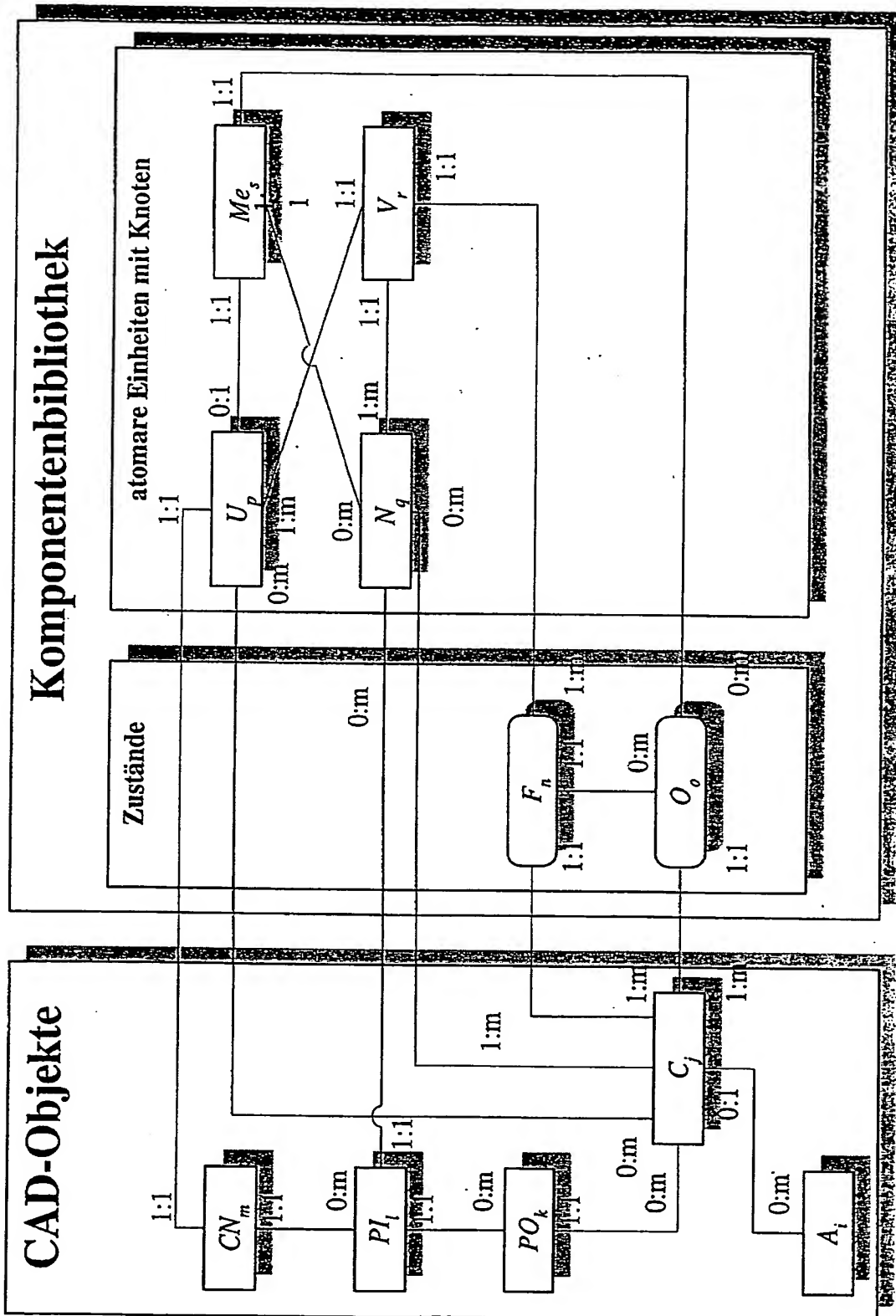


Fig. 2

Komponente	elektrische Zustände	Betriebszustände	Komponentenzustände (Komponentenfehler)
Widerstand	1: $R_{12} = R_n$		1: O.K.
	2: $R_{12} = \infty$		2: defekt
Schalter	1: $R_{12} = 0$	1: geschlossen	1: O.K.
	1: $R_{12} = 0$	2: offen	3: offen und Kontakt
	2: $R_{12} = \infty$	1: geschlossen	2: geschlossen und kein Kontakt
	2: $R_{12} = \infty$	2: offen	1: O.K.
Leitung	1: $R_{12} = 0$		1: O.K.
	2: $R_{12} = \infty$		2: Unterbrechung
Lampe	1: $R_{12} = R_n$	1: Lampe EIN $I_{12} = I_n$	1: O.K.
	1: $R_{12} = R_n$	2: Lampe AUS $I_{12} < I_n$	1: O.K.
	2: $R_{12} = \infty$	2: Lampe AUS $I_{12} < I_n$	2: defekt
Leitungsbündel	1.1: $R_{1x-2x} = 0 \wedge$		1: O.K.
	2.2: $R_{ax-ay} = \infty$		
	1.2: $R_{1x-2x} = \infty$		2: Leitungsunterbrechung
	2.1: $R_{ax-ay} = 0$		3: Leitungskurzschluss
Stecker	1.1: $R_{1x-2x} = 0 \wedge$		1: Steckverbinder O.K.
	2.2: $R_{1x-1y} = \infty$		
	1.2: $R_{1x-2x} = \infty$		2 Steckverbinder offen
	2.1: $R_{1x-1y} = 0$		3: Kurzschluss der Pins
Spannungsquelle	1: $V_i = V_n$		1: Spannungsquelle O.K.
	2: $V_i = 0$		2: Spannungsquelle defekt

Fig. 3a

Komponente	elektrische Zustände	Betriebszustände	Komponentenzustände (Komponentenfehler)
Relais	1.1: $R_1 = 0$	1: Relaiskontakt aktiv $I_2 = I_n$	1.1: Relaiskontakte O.K.
	2.1: $R_2 = R_n$		2.1: Relaiswicklung O.K.
	1.2: $R_1 = \infty$	1: Relaiskontakt aktiv $I_2 = I_n$	1.1: Relaiskontakte defekt
	2.1: $R_2 = R_n$		2.1: Relaiswicklung O.K.
	1.2: $R_1 = \infty$	2: Relaiskontakt passiv $I_2 < I_n$	1.1: Relaiskontakte defekt
	2.2: $R_2 = \infty$		2.2: Relaiswicklung unterbrochen
Sicherung	1.1: $R_1 = 0$	2: Relaiskontakt passiv $I_2 < I_n$	1.1: Relaiskontakte O.K.
	2.2: $R_2 = \infty$		2.2: Relaiswicklung unterbrochen
Sicherung	1: $R_{12} = 0$		1: Sicherung O.K.
	2: $R_{12} = \infty$		2: Sicherung defekt
Magnetventil	1: $R_{12} = R_n$		1: Magnetventil O.K.
	2: $R_{12} = \infty$		2: Magnetventil defekt
Motor	1: $R_{12} = R_n$		1: Motor O.K.
	2: $R_{12} = R_w$		2: Rotor blockiert
	3: $R_{12} = \infty$		3: Wicklungsunterbrechung
Signalgeber	1: $R_{12} = R_n$		1: Signalgeber O.K.
	2: $R_{12} = \infty$		2: Signalgeber defekt
Verbinder	1: $R_i = 0$		1: Verbindung O.K.
	2: $R_i = \infty$		2: Verbindung unterbrochen

Fig. 3b



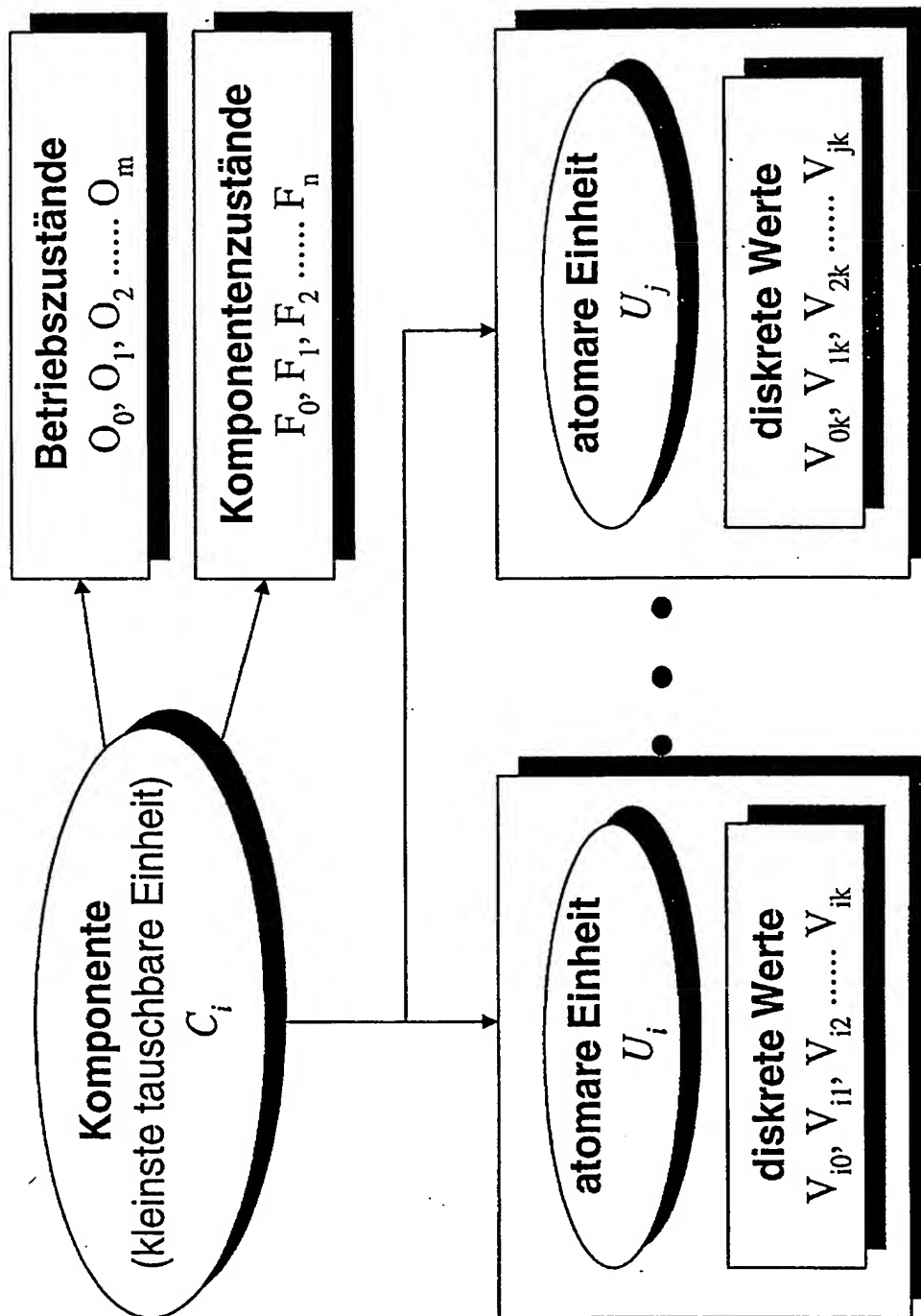


Fig. 4

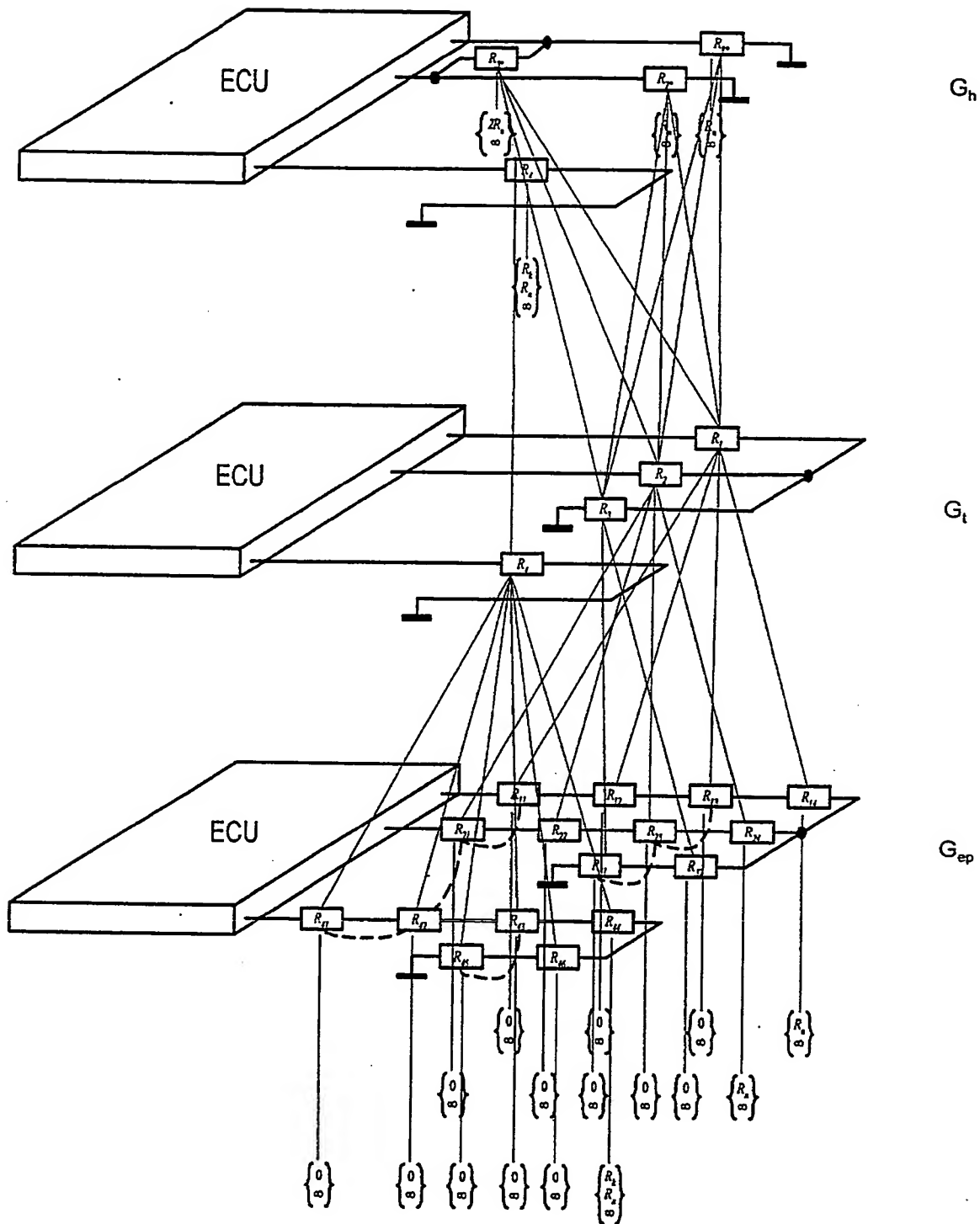


Fig. 5

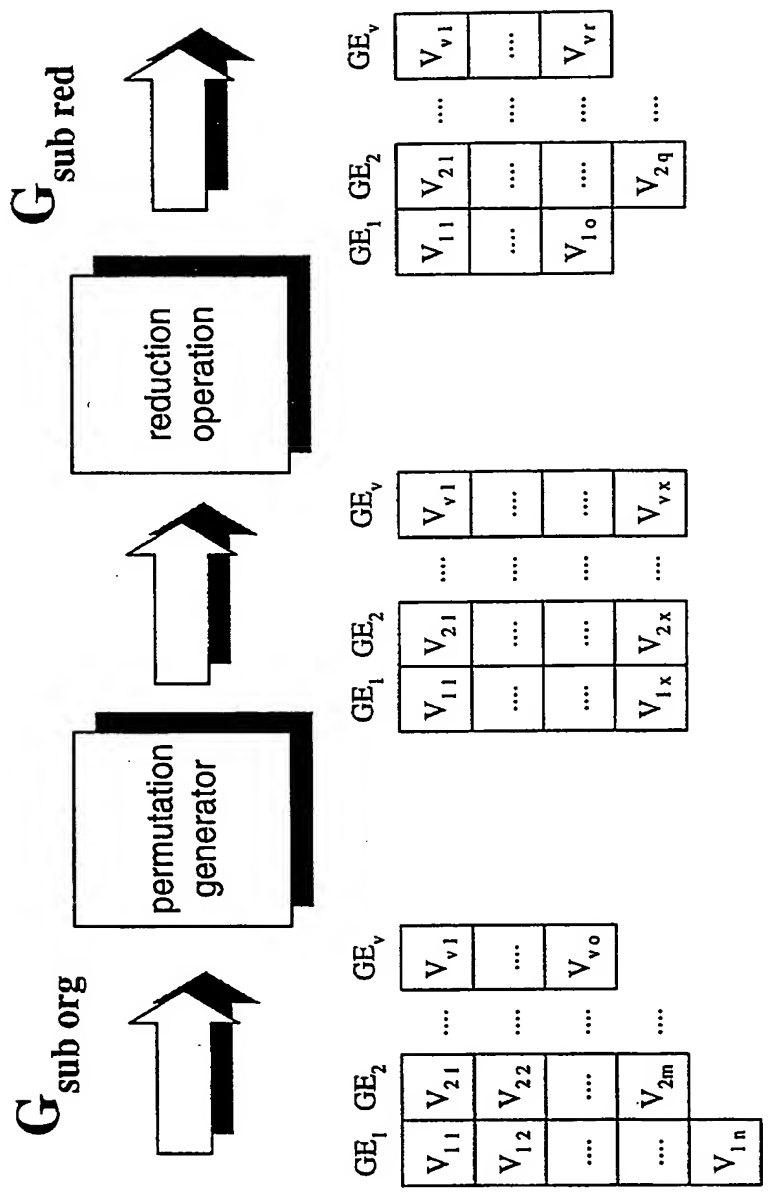


Fig. 6

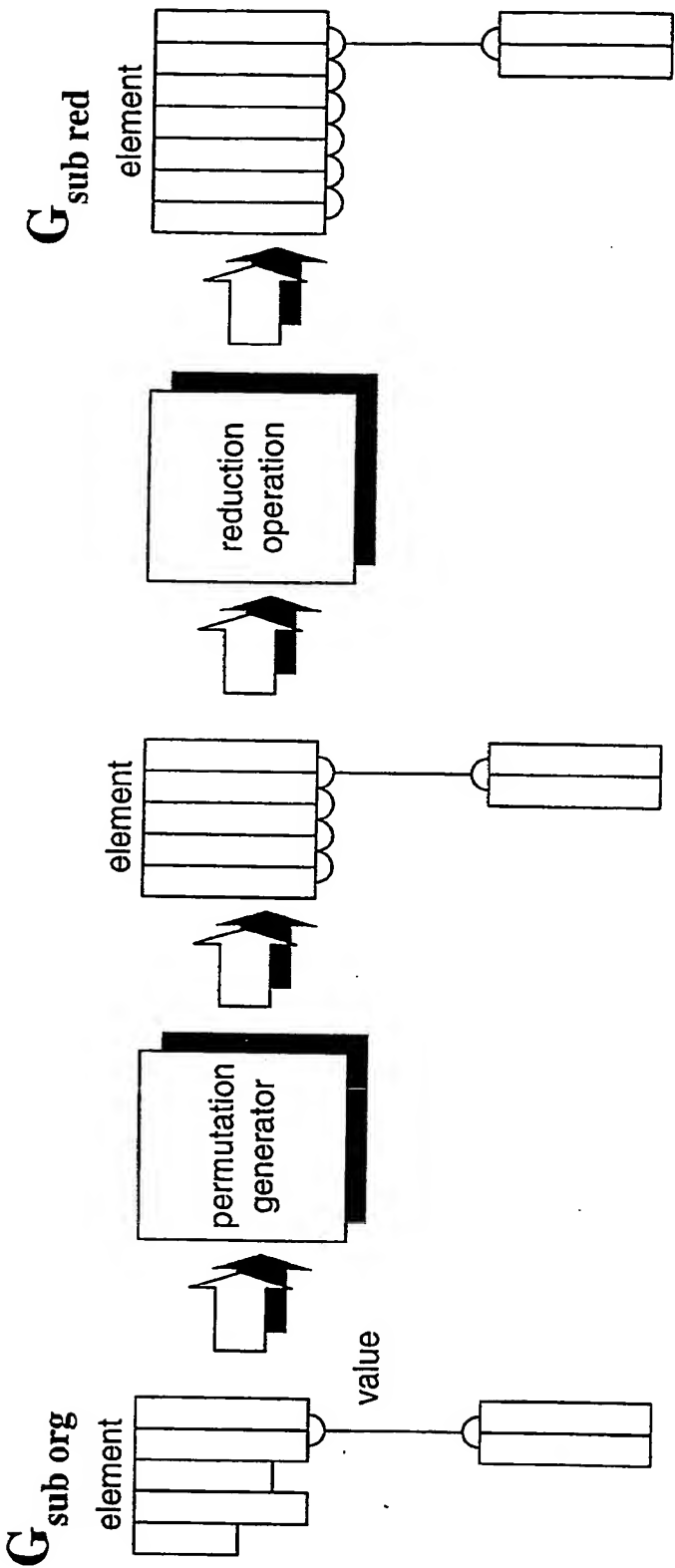


Fig. 7

Komponenten mit deren Komponentenzuständen				Werte der Grundbausteine des Kopfgraphen			
$wa$	$wb$	$Ba$	....	$R_1$	$R_2$	$R_3$	....
....	....	....		....	....	....	
0	0	0		$R_n$	$R_n$	$\infty$	
0	0	1	....	$\infty$	$R_n$	$\infty$	....
1	0	1		$\infty$	$R_n$	$\infty$	
....	....	....	....	....	....	....	....
0	1	0		$R_n$	$\infty$	$\infty$	
....	....	....					

Fig.8

Fig. 9a

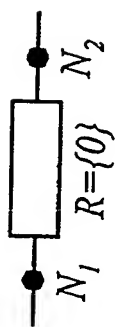
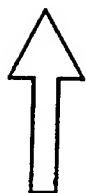
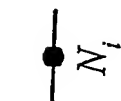


Fig. 9b



Fig. 9c

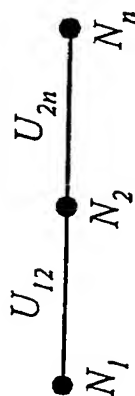
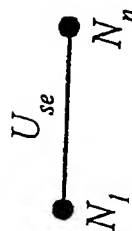
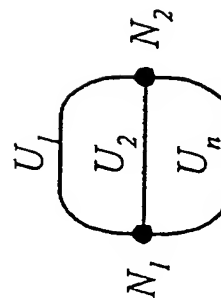
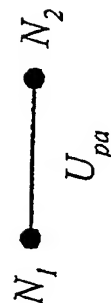


Fig. 9d



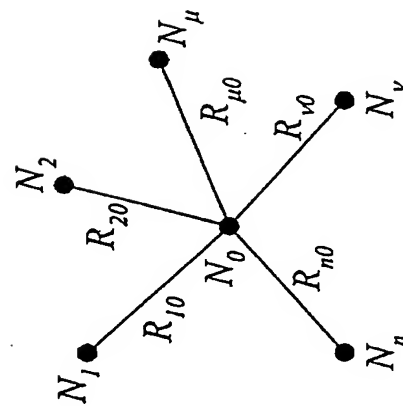
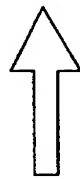
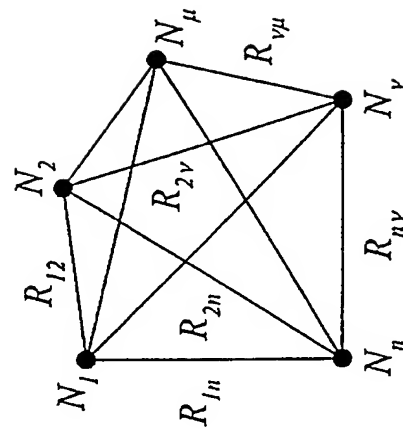


Fig. 10

Fig. 11a

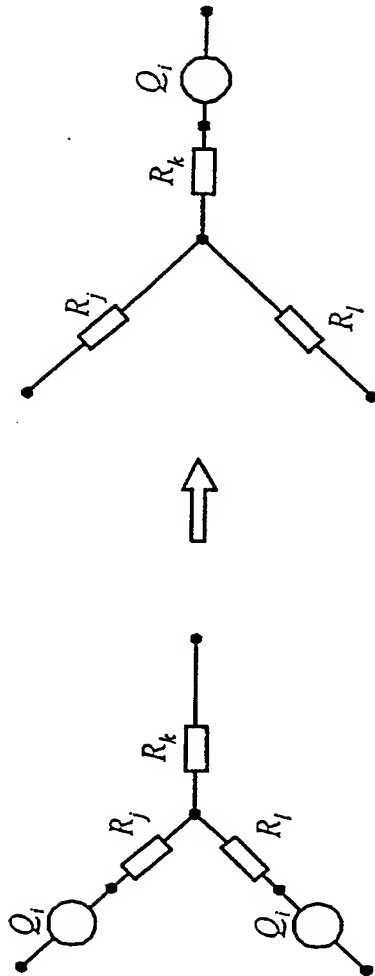


Fig. 11b

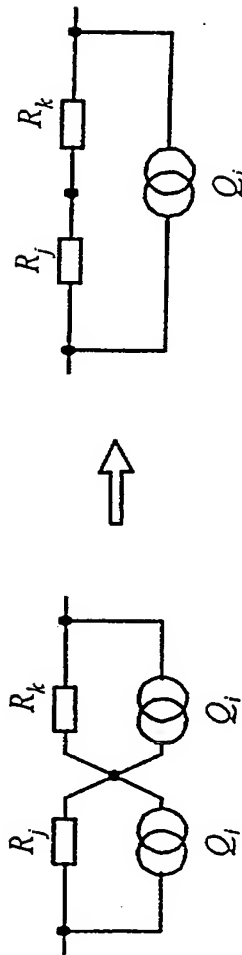


Fig. 11c

